

THE LECTURER: Toby Riley is a C-E-I-R, INC., senior consultant in program development who has more than ten years experience in the field of computer sciences. He has served as project director for such large-scale programs as Du Pont's RAMPS



(8,500 FORTAN statements) and the NASA Launch Vehicle Model. Riley is a member of ACM and the British Computer Society, and has been guest lecturer in computer sciences at the London School of Economics, the Regent Street Polytechnic and Woolwich Polytechnic Institute. In addition, he has conducted programming courses for more than 200 scientists and 50 professional programmers for NASA. He is currently conducting similar courses for the FAA and the USDA Graduate School. During his experience with both IBM and C-E-I-R, Riley has programmed on all the IBM 1400 and 7000 series computers, as well as several systems in the UNIVAC Scientific series, and Burroughs, GE, and RCA computers. Riley received his undergraduate degree from Swarthmore and has completed two years of graduate studies at the Case Institute of Technology.



REGISTRATION: Tuition, including noon meals and all course materials, is \$220 for the first student and \$195 for each additional student from the same organization. Classes begin each day at 9 a.m. and continue through 5 p.m. Hotel rooms are not included.

HOTEL ACCOMMODATIONS: The seminar will be given at the Statler Hilton Hotel, 16th and K Streets, N.W., Washington, D. C. The Statler Hilton will hold a block of rooms for advance reservation by students until two weeks prior to the seminar.

OTHER COURSES: Among other seminars in the C-E-I-R seminar program is:

SURVEY OF DATA COMMUNICATIONS

For data processing personnel preparing for "on line" or "real time" systems involving data communications. Covers hardware and software considerations important to successful system design, and data communications concepts and terminology. Executive House, Washington, D. C., June 21-22-23.

For an outline of this seminar or others scheduled in coming months, write:

Registrar

Institute for Advanced Technology

C-E-I-R^{INC}

**1200 Jefferson Davis Highway
Arlington, Virginia 22202**

COMPUTER PROGRAM DOCUMENTATION & DEBUGGING

FIRST DAY

DOCUMENTATION CONCEPTS

Documentation Consumers

Identification of the Consumers of Documentation. Requirements and variable needs of programmers, operators, management and users.

Elements of Documentation

Taxonomy and definitions of tools and artifacts of documentation. Alternative means of documenting the same information.

Current Forms of Documentation

Review of various types of forms and their application.

Automated Documentation

Discussion and explanation of several available automated block-diagramming and documentation program systems. Problem exercise.

Evening Workshop Session

"Hands-on" computer processing of documentation exercise at C-E-I-R Computer Center.

SECOND DAY

STANDARDS AND PRACTICES

Program Documentation Standards

Review of standards development at the organizational, national, and international level. Eg. American Standards Association standard on Flow-Chart Symbols.

Current Practices in Documentation

Review and critique of documentation procedures and standards used by several organizations.

Documentation Support Activities

Presentation of the services provided by a staff support group in documentation.

Problems of Documentation

Evaluation of implementation problems for

maintaining valid documentation, and discussion for their solution.

Case Study in Documentation Practices

Field trip to a local installation to review documentation concepts, procedures, and standards.

THIRD DAY

PROGRAM DEBUGGING

Definition and Scope of Debugging

Defining means for determining when a program is completed. Discussion of error-free coding and its cost.

Types of Errors

Taxonomy of various types of errors, and their authors.

Software Aids for Debugging

Review of commonly used techniques for error detection and removal. A detailed description of several new software systems.

FOURTH DAY

PROGRAM PLANNING

Causes of Errors

Outline of reasons why errors occur, by level of organization and their effects. Applications, pressures, attitudes, etc.

Preventative Measures Versus Cures

Discussion of methods for preventing errors. Selection of coding language, program versus problem segmentation.

Effects of Programming Environment on Debugging

Presentation of techniques for creating environment for programming efficiency, and related procedural practices.

Summary and Conclusion

Open discussion and review, questions and answers.

C-E-I-R^{INC.} is an international applied research and data processing corporation that offers analytical, scientific and computer services to business, science and government. Founded in 1954, it is today the world's largest, most experienced and best equipped independent organization in its field. The C-E-I-R professional staff includes several hundred mathematicians, statisticians, economists, operations researchers, management scientists and others from a variety of disciplines. In addition, many of the finest scientific and professional men in America are retained on a consultant basis. Augmenting this professional capability are modern electronic computing equipment, and skilled computer programmers and operations personnel at computing centers in five major U.S. cities, London, The Hague and Mexico City.



is the latest expression of C-E-I-R's long-standing involvement in management education. The revolutionary nature of computer-based methods made education an integral part of C-E-I-R operations from the beginning. This relationship is formalized through the Institute for Advanced Technology. IAT faculty members are drawn primarily from the ranks of C-E-I-R's professional and computer operations staffs. Its curriculum is drawn from subjects in which C-E-I-R staff members are expert—recognized for excellence in day-to-day application of the art and science of computer usage to the real problems of C-E-I-R customers in business and industry. Sharing these new skills through seminars in major cities, management clinics, and special, in-plant training programs is the major goal of the Institute for Advanced Technology.

INSTITUTE FOR ADVANCED TECHNOLOGY

1200 JEFFERSON DAVIS HIGHWAY • ARLINGTON, VIRGINIA 22202 • (703) 684-6377

DOCUMENTATION AND
DEBUGGING -- BASICS OF
EFFECTIVE PROGRAMMING

Dear Sir:

You may want to argue with Toby Riley during his presentation of the Computer Program Documentation & Debugging seminar. As a matter of fact, we hope you will, because the ground he covers involves two controversial aspects of programming that are not often approached in terms of objective standards.

Riley has some firm beliefs about documentation and debugging acquired over a ten-year career in programming and programmer management. Working for a company that built its reputation largely in development of computer software on a contract basis, Riley's beliefs are, frankly, colored by the need to deliver computer programs on time, to specifications, and within a budget.

If you have seen programming money slurp down the drain because of inadequate (or over-adequate) documentation, or through debugging that cost too much man time and machine time, you may have considered such possible causes as:

- Informal or non-existent training in two basic programming skills--documentation and debugging.
- The absence of practical standards for defining and managing the documentation and debugging tasks.

That's what this seminar is concerned with. Designed for both the programmer and the programming manager, it covers documentation and debugging from A to Z at a practical, working level. In each area it first defines the subject, then reviews current practices, and winds up with an outline for workable performance standards.

You may have concluded at this point that Riley figures he knows it all. He doesn't. And the conclusions reached in this seminar won't necessarily be his own--he expects you to come prepared to argue the point whenever you think he's off the track. The idea is to open up discussion and lead it to a consensus acceptable to the entire group.

We said that a good part of C-E-I-R's reputation is built on its performance in computer software development. And that Riley, in his years with IBM and C-E-I-R has developed a healthy appreciation and understanding of the importance of documentation and debugging.

An example of what we mean is C-E-I-R's development of a resource allocation model (under the lecturer's direction) with the E.I. du Pont de Nemours & Company. This program consists of more than 8,500 FORTRAN statements, and was under development for two years. Throughout this period, programmers, mathematicians, and engineers were kept in constant communication regarding development of algorithms, practicality of the approach, and data processing problems. The program was subjected to in-house testing, then on-site field testing in several different client locations, each with different conditions.

In every instance, the documentation had to be complete for each of the different staffs involved during the year the program underwent field tests. This documentation included program lists, explanation of algorithms, test correspondence, training texts and user's guides. Tasks such as this one helped C-E-I-R (and Riley) appreciate the problems of programming for multi-installation use, and meeting a variety of specifications for a common program.

During this seminar you will be asked to consider the problem of meaningful problem definition, and answer the question of what to document, and when to document it, and why.

Each participant is asked to bring along two computer programs: one that you believe to represent most features of a well-documented program, and a second program that you will use in a class exercise.

In the class problem you will see your second program processed by two automatic documentation systems [Flowcharter and the Symbolic Block Diagram System] at C-E-I-R's computing center. You will discuss and evaluate the results of these systems, and consider other special documenting routines that can be written or used by programmers.

This seminar will benefit any practicing programmer or programmer manager by focusing his attention on important aspects of his job whose proper handling can:

- o Extend program life by simplifying the job of making changes and conversions.
- o Improve manager-programmer communication by setting ground rules for objectives, methods and status reporting.
- o Facilitate the exchange of information between programmers and thereby better coordinate the group effort.
- o Reduce the amount of man and machine time spent in program debugging.
- o Increase the efficiency--and reduce the cost -- of the programming operation.

This seminar will be given June 14, 15, 16 and 17 at the Statler-Hilton Hotel in Washington, D.C. These are four days that can help you advance your professional career. Circle them on your calendar now, and mail the enclosed registration form. We look forward to meeting you.

Sincerely,


Robert D. Nixon
Curriculum Director

p.s. If the June date is inconvenient for you, please ask about future sessions. Because class sizes are limited, additional unannounced presentations of C-E-I-R seminars are often added to the schedule.

In every instance, the documentation had to be complete for each of the different staffs involved during the year the program underwent field tests. This documentation included program lists, explanation of algorithms, test correspondence, training texts and user's guides. Tasks such as this one helped C-E-I-R (and Riley) appreciate the problems of programming for multi-installation use, and meeting a variety of specifications for a common program.

During this seminar you will be asked to consider the problem of meaningful problem definition, and answer the question of what to document, and when to document it, and why.

Each participant is asked to bring along two computer programs: one that you believe to represent most features of a well-documented program, and a second program that you will use in a class exercise.

In the class problem you will see your second program processed by two automatic documentation systems [Flowcharter and the Symbolic Block Diagram System] at C-E-I-R's computing center. You will discuss and evaluate the results of these systems, and consider other special documenting routines that can be written or used by programmers.

This seminar will benefit any practicing programmer or programmer manager by focusing his attention on important aspects of his job whose proper handling can:

- o Extend program life by simplifying the job of making changes and conversions.
- o Improve manager-programmer communication by setting ground rules for objectives, methods and status reporting.
- o Facilitate the exchange of information between programmers and thereby better coordinate the group effort.
- o Reduce the amount of man and machine time spent in program debugging.
- o Increase the efficiency--and reduce the cost -- of the programming operation.

This seminar will be given June 14, 15, 16 and 17 at the Statler-Hilton Hotel in Washington, D.C. These are four days that can help you advance your professional career. Circle them on your calendar now, and mail the enclosed registration form. We look forward to meeting you.

Sincerely,


Robert D. Nixon
Curriculum Director

p.s. If the June date is inconvenient for you, please ask about future sessions. Because class sizes are limited, additional unannounced presentations of C-E-I-R seminars are often added to the schedule.